What is claimed is:

1. A buffer for storing information organized into packets, comprising:

a destructive-read dynamic random access memory (drDRAM), information stored in the drDRAM being destroyed following the information being read therefrom;

a random access memory (RAM);

at least one queue for storing a number of pointers to locations where packets are stored in the drDRAM and the RAM, wherein incoming fanout splitting packets are stored in the RAM and remaining incoming packets are stored in the drDRAM, and the (head-of-line) packet pointed to by the pointer at the top of the at least one queue being automatically read from the drDRAM and stored in the RAM for subsequently providing the packet externally to the buffer.

2. The buffer of claim 1, wherein the RAM is a static random access memory (SRAM).

3. The buffer of claim 1, wherein the RAM is a dynamic random access memory (DRAM), information stored in the DRAM

is not destroyed following the information being read therefrom.

4. The buffer of claim 1, wherein the at least one queue comprises a plurality of queues, each queue capable of storing a number of pointers to locations where received packets are stored in the drDRAM and RAM, the head-of-line packet pointed to by the pointer at the top of each queue being automatically read from the drDRAM and stored in the RAM for subsequently providing the packet externally to the buffer.

5. The buffer of claim 1, wherein an incoming packet to the buffer is automatically stored in the RAM in the event the at least one queue is empty.

6. The buffer of claim 1, wherein following the head-of-line packet being read from the buffer, a new head-of-line packet is read from the drDRAM and stored in the RAM if the at least one queue is not empty.

7. The buffer of claim 1, wherein the drDRAM is larger than the RAM.

8. The buffer of claim 1, wherein the drDRAM is organized in rows and columns of memory cells, at least one row being divided into a plurality of groups of memory cells, each group of memory cells being selectively individually addressable.

9. The buffer of claim 1, wherein incoming unicast packets or nonfanout splitting packets are initially automatically stored in the drDRAM.

10. A method of storing packets of information, comprising:

receiving a packet of information;

identifying a queue to which the received packet is to be associated;

determining whether the identified queue is empty; and

storing the received packet in at least one of a first memory and a second memory based upon the determination of whether the queue is empty.

11. The method of claim 10, further comprising:

determining whether the received packet is a fanout splitting packet, wherein the step of storing is based upon the determination of whether the received packet is a fanout splitting packet.

12. The method of claim 11, wherein the step of storing comprises storing the received packet in the first memory if the received packet is determined not to be a fanout splitting packet and storing the received packet in the

second memory if the received packet is determined to be a fanout splitting packet.

13. The method of claim 10, further comprising:

determining whether the received packet is a unicast packet or a nonfanout splitting packet, wherein the step of storing is based upon the determination of whether the received packet is a unicast packet or a nonfanout splitting packet.

14. The method of claim 10, wherein the received packet is stored in the first memory based upon a determination that the queue is not empty, and stored in the second memory based upon a determination that the queue is empty.

15. The method of claim 14, further comprising transferring the received packet from the first memory to the second memory upon the received packet becoming the head-of-line packet associated with the queue.

16. A method of handling packets of information, comprising:

storing a first incoming packet of information in a first memory, the first packet being associated with at least one queue;

transferring the first packet from a first memory to a second memory upon the first packet becoming the head-of-line packet for the at least one queue; and

sending the first packet to a telecommunications device following the step of transferring.

17. The method of claim 16, further comprising:

determining whether a second incoming packet is a fanout splitting packet; and

storing the second incoming packet in one of the first memory and the second memory based upon the step of determining.

18. The method of claim 17, wherein the step of storing the second incoming packet comprises storing the second incoming packet in the first memory upon the affirmative

determination that the second incoming packet is not a fanout splitting packet, and storing the second incoming packet in the second memory upon the affirmative determination that the second incoming packet is a fanout splitting packet.

19. The method of claim 16, further comprising:

receiving a second incoming packet;

identifying a queue for containing a pointer for pointing to a location where the second incoming packet is to be stored;

determining whether the identified queue is empty; and

storing the second incoming packet in one of the first memory and the second memory based upon the step of determining.

20. The method of claim 16, further comprising:

receiving a second packet of information;

determining whether the at least one queue is empty; and

storing the second packet in the second memory based upon the step of determining.

21. A system, comprising:

a node for communicating packets of information, including a data buffer for receiving incoming packets of information and selectively storing the packets in a first memory, the data buffer including at least one queue for pointing to locations in the data buffer where the packets are stored, wherein a first packet is transferred from the first memory to a second memory upon the first packet becoming the head-of-line packet for the at least one queue, the node transmitting the first packet from the second memory externally to the node following the first packet being stored in the second memory.

22. The system of claim 21, wherein the first memory comprises a read-once memory in which stored data is destroyed following being read from the first memory a first time.

23. The system of claim 22, wherein the second memory comprises a static random access memory.

24. The system of claim 22, wherein the second memory comprises a dynamic random access memory which performs nondestructive memory read operations.

25. The system of claim 21, wherein an incoming packet is initially stored in the second memory in the event the at least one queue is empty.